数理解析研究所講究録 1366 巻 2004 年 91-99

Persistence of Termination for Non-Overlapping Term Rewriting Systems

岩見 宗弘 (Munehiro Iwami)
島根大学・総合理工学部
Faculty of Science and Engineering, Shimane University
e-mail: munehiro@cis.shimane-u.ac.jp

Abstract

A property P is called persistent if for any many-sorted term rewriting system $\mathcal{R}$, $\mathcal{R}$ has the property P if and only if term rewriting system $\Theta(\mathcal{R})$, which results from $\mathcal{R}$ by omitting its sort information, has the property P. In this paper, we show that termination is persistent for non-overlapping term rewriting systems and we give example as application of this result. Furthermore we obtain that completeness is persistent for non-overlapping term rewriting systems.
keywords: term rewriting system, termination, persistence, non-overlapping, weak innermost normalization, completeness

1 Introduction

Term rewriting systems (TRSs) can offer both flexible computing and effective reasoning with equations. TRSs have been widely used as a model of functional and logic programming languages and as a basis of theorem provers, symbolic computation, algebraic specification and software verification [3, 4, 7, 10].

A rewrite system is called *terminating (strongly normalizing)* if there exists no infinite reduction sequence. In a *confluent* rewrite system, the normal form of a given term is unique, that is, the final result does not depend on the strategy in which the rewrite rules were applied. Termination and confluence are the fundamental properties of TRSs. It is well-known that termination and confluence are undecidable for TRSs in general [3, 5].

Zantema [13] introduced the notion of *persistence* as follows: A property P is called persistent if for any many-sorted TRS $\mathcal{R}$, $\mathcal{R}$ has the property P if and only if TRS $\Theta(\mathcal{R})$, which results from $\mathcal{R}$ by omitting its sort information, has the property P. Zantema [13] showed that termination is persistent for TRSs without collapsing or duplicating rules. However termination is *not* persistent in general [13]. The basic counterexample from Toyama [12] leads to the following sorted TRS $\mathcal{R}$:

$$\mathcal{R} = \begin{cases} f(0,1,x) \to f(x,x,x) \\ g(y,z) \to y \\ g(y,z) \to z \end{cases}$$

where the set of sorts $\mathcal{S} = \{\alpha, \beta\}$ and the function symbols and variables are defined as follows:

$$f : \alpha \times \alpha \times \alpha \to \alpha, 0 : \alpha, 1 : \alpha, g : \beta \times \beta \to \beta, x : \alpha, y : \beta, z : \beta$$

The sorted TRS $\mathcal{R}$ is terminating. Let Θ be a sort elimination function. Then underlying TRS $\Theta(\mathcal{R})$, which results from $\mathcal{R}$ by omitting its sort information, is not terminating.

$$\begin{aligned} f(\underline{g(0,1)}, g(0,1), g(0,1)) &\to_{\Theta(\mathcal{R})} f(0, \underline{g(0,1)}, g(0,1)) \\ &\to_{\Theta(\mathcal{R})} \underline{f(0, 1, g(0,1))} \\ &\to_{\Theta(\mathcal{R})} f(\underline{g(0,1)}, g(0,1), g(0,1)) \\ &\to_{\Theta(\mathcal{R})} \cdots \end{aligned}$$

is an infinite reduction in $\Theta(\mathcal{R})$. In each step the contracted *redex* is underlined. Aoto and Toyama showed the persistence of confluence [1] and usual many-sorted TRS was extended with ordered sorts [2]. It was shown that the persistence of confluence is preserved for this extension in [2]. Ohsaki and Middeldorp [11] studied the persistence of termination, acyclicity and non-loopingness on equational many-sorted TRSs.

In this paper, we show the persistence of termination for non-overlapping TRSs and we give the example as application of this result. Zantema's result can *not* be applied to our example. Our result provides a new and powerful tool for proving termination of TRSs. As a result we obtain the persistence of completeness for non-overlapping TRSs.

In section 2, many-sorted TRS is formulated and well-sortedness is characterized in section 3. First, we show the persistence of weak innermost normalization. Next, we show the persistence of termination for non-overlapping TRSs and we give the example as an application of this result in section 4. Furthermore, we obtain the persistence of completeness for non-overlapping TRSs.

2 Preliminaries

We mainly follow basic definitions in the literature [1, 7].

2.1 Sorted term rewriting systems

In this subsection, we introduce the basic notions of sorted term rewriting systems. Usual term rewriting systems [3] are considered as special cases of sorted term rewriting systems.

Let $\mathcal{S}$ be a set of *sorts* and $\mathcal{V}$ be a set of countably infinite *sorted variables*. We assume that there are countably infinite variables of sort α for each sort $\alpha \in \mathcal{S}$. Let $\mathcal{F}$ be a set of sorted function symbols. We assume that each sorted function symbol $f \in \mathcal{F}$ is given with the sorts of its arguments and the sort of its value, all of which are included in $\mathcal{S}$. We write $f{:}\alpha_1 \times \ldots \times \alpha_n \to \beta$ if and only if f takes n arguments of sorts $\alpha_1, \ldots, \alpha_n$ respectively to a value of sort β. Function symbol of with no arguments is *constant.*

The set $\mathcal{T}(\mathcal{F}, \mathcal{V}) = \cup_{\alpha \in \mathcal{S}} \mathcal{T}(\mathcal{F}, \mathcal{V})^\alpha$ of all *sorted terms* built from $\mathcal{F}$ and $\mathcal{V}$ is defined as follows: (1) $\mathcal{V}^\alpha \subseteq \mathcal{T}(\mathcal{F}, \mathcal{V})^\alpha$, (2) $f{:}\alpha_1 \times \ldots \times \alpha_n \to \alpha$, $t_i \in \mathcal{T}(\mathcal{F}, \mathcal{V})^{\alpha_i}$ $(i = 1, \ldots, n)$

then $f(t_1,\ldots,t_n) \in \mathcal{T}(\mathcal{F},\mathcal{V})^\alpha$. Here $\mathcal{T}(\mathcal{F},\mathcal{V})^\alpha$ denotes the set of all sorted terms of sort α.

We write $t : \alpha$ if t is of sort α. $\mathcal{V}(t)$ denotes the set of all variables in t. $\mathcal{T}(\mathcal{F},\mathcal{V})^\alpha$ and $\mathcal{T}(\mathcal{F},\mathcal{V})$ are abbreviated as $\mathcal{T}^\alpha$ and $\mathcal{T}$, respectively. Let $\Box^\alpha$ be a special constant (*hole*) of sort α. Elements of $\mathcal{T}(\mathcal{F} \cup \{\Box^\alpha \mid \alpha \in \mathcal{S}\},\mathcal{V})$ are called *contexts* over $\mathcal{T}(\mathcal{F},\mathcal{V})$. We write $C{:}\alpha_1 \times \ldots \times \alpha_n \to \alpha$ if and only if the sort of context C is α and it has n holes $\Box^{\alpha_1},\ldots,\Box^{\alpha_n}$. If $C{:}\alpha_1 \times \ldots \times \alpha_n \to \alpha$ and $t_i{:}\alpha_i$ $(i = 1,\ldots,n)$ then $C[t_1,\ldots,t_n]$ denotes the term obtained from C by replacing holes with $t_1,\ldots,t_n$ from left to right. A context that contains precisely one hole is denoted by $C[\,]$. A term t is said to be a *subterm* of s if and only if $s = C[t]$ for some context C. A *substitution* θ is a mapping from $\mathcal{V}$ to $\mathcal{T}$ such that $x \in \mathcal{V}^\alpha$ implies $\theta(x) \in \mathcal{T}^\alpha$. A substitutions over terms is defined as a homomorphic extension. $\theta(t)$ is usually written as $t\theta$. A *sorted rewrite rule* on $\mathcal{T}$ is a pair $l \to r$ such that $l \notin \mathcal{V}$, $\mathcal{V}(r) \subseteq \mathcal{V}(l)$, sorted terms l and r have the same sort. A *sorted term rewriting system* (STRS, for short) is a pair $(\mathcal{F},\mathcal{R})$ where $\mathcal{F}$ is a set of sorted function symbols and $\mathcal{R}$ is a set of sorted rewrite rules on $\mathcal{T}(\mathcal{F},\mathcal{R})$. $(\mathcal{F},\mathcal{R})$ is often abbreviated as $\mathcal{R}$ and in that case $\mathcal{F}$ is defined to be the set of function symbols that appear in $\mathcal{R}$.

Given a STRS $\mathcal{R}$, a sorted term s is *reduced* to a sorted term t ($s \to_{\mathcal{R}} t$, in symbol) when $s = C[l\theta]$ and $t = C[r\theta]$ for some rewrite rule $l \to r \in \mathcal{R}$, context C and substitution θ. We call $s \to_{\mathcal{R}} t$ a *rewrite step* or *reduction* from s to t of $\mathcal{R}$. $l\theta$ is called *redex* of this rewrite step. One can easily check that sorted terms s and t have the same sort whenever $s \to_{\mathcal{R}} t$.

The transitive reflexive closure of $\to_{\mathcal{R}}$ is denoted by $\to^*_{\mathcal{R}}$. Terms t_1 and t_2 are *joinable* if there exists some term t' such that $t_1 \to^*_{\mathcal{R}} t' \leftarrow^*_{\mathcal{R}} t_2$. A term t is *confluent* if for any terms t_1 and t_2, t_1 and t_2 are joinable whenever $t_1 \leftarrow^*_{\mathcal{R}} t \to^*_{\mathcal{R}} t_2$. A STRS $\mathcal{R}$ is confluent if every term is confluent to $\to_{\mathcal{R}}$. A term t is a *normal form* if there is no term t' such that $t \to_{\mathcal{R}} t'$. A term t is *terminating (strongly normalizing)* if there is no infinite reduction sequence starting from term t. A STRS $\mathcal{R}$ is terminating if every term is terminating to $\to_{\mathcal{R}}$. A STRS $\mathcal{R}$ is *weakly innermost normalizing* if every term has a normal form which can be reached by an *innermost* reduction. In an innermost reduction a redex may only be contracted if it contains no proper subredexes. In that case we write $s \to_{i\mathcal{R}} t$. A STRS $\mathcal{R}$ is *complete* if $\mathcal{R}$ is confluent and terminating. Every terminating STRS is weakly innermost normalizing.

A rewrite rule $l \to r$ is a *collapsing* rule if r is a variable. A rewrite rule $l \to r$ is a *duplicating* rule if some variable has more occurrences in r than in l. Let $l_1 \to r_1$ and $l_2 \to r_2$ be renamed versions of rewrite rules in a STRS $\mathcal{R}$ such that they have no variables in common. Suppose $l_1 = C[t]$ with $t \notin \mathcal{V}$ such that t and l_2 are unifiable, i.e. $t\theta = l_2\theta$ for a most general unifier θ. The term $l_1\theta = C[l_2]\theta$ is subject to the rewrite steps $l_1\theta \to_{\mathcal{R}} r_1\theta$ and $l_1\theta \to_{\mathcal{R}} C[r_2]\theta$. Then the pair of reducts $\langle C[r_2]\theta, r_1\theta\rangle$ is called a *critical pair* of $\mathcal{R}$. A STRS $\mathcal{R}$ is said to be *non-overlapping* if there is no critical pair between rules of $\mathcal{R}$.

When $\mathcal{S} = \{*\}$, an STRS is called a term rewriting system (TRS, for short). Given an arbitrary STRS $\mathcal{R}$, by identifying each sort with $*$, we obviously obtain a TRS $\Theta(\mathcal{R})$ - called the underlying TRS of $\mathcal{R}$.

2.2 Sorting of term rewriting systems

Aoto and Toyama [1] defined the notion of *sort attachment* and formulated the notion of persistence using sort attachment. We mainly follow basic definitions in [1] in this subsection.

Let $\mathcal{F}$ and $\mathcal{V}$ be sets of function symbols and variables, respectively, on a trivial set $\{*\}$ of sorts. Terms built from this language are called *unsorted terms.* Let $\mathcal{S}$ be another set of sorts. A *sort attachment* τ *on* $\mathcal{S}$ is a mapping from $\mathcal{F} \cup \mathcal{V}$ to the set $\mathcal{S}^*$ of finite sequences of elements from $\mathcal{S}$ such that $\tau(x) \in \mathcal{S}$ for any $x \in \mathcal{V}$ and $\tau(f) \in \mathcal{S}^{n+1}$ for any n-ary function symbol $f \in \mathcal{F}$. We write $\tau(f) = \alpha_1 \times \ldots \times \alpha_n \to \beta$. Without loss of generality we assume that there are countably infinite variables x with $\tau(x) = \alpha$ for each $\alpha \in \mathcal{S}$. The set of τ-sorted function symbols from $\mathcal{F}$ is denoted by $\mathcal{F}^\tau$.

A term t is said to be *well-sorted* under τ with sort α if $t : \alpha$ is derivable in the following rules: (1) $\tau(x) = \alpha$ implies $x{:}\alpha$, (2) $\tau(f) = \alpha_1 \times \ldots \times \alpha_n \to \beta$, $t_1{:}\alpha_1, \ldots, t_n{:}\alpha_n$ imply $f(t_1, \ldots, t_n){:}\beta$.

The set of well-sorted terms under τ is denoted by $\mathcal{T}^\tau$, i.e. $\mathcal{T}^\tau = \{t \in \mathcal{T} \,|\, t : \alpha$ for some $\alpha \in \mathcal{S}\}$. Clearly, $\mathcal{T}^\tau \subseteq \mathcal{T}$. For a context C, we write $C{:}\alpha_1 \times \ldots \times \alpha_n \to \beta$ if $C[\Box^{\alpha_1}, \ldots, \Box^{\alpha_n}]{:}\beta$ is derivable by rules (1), (2) with an additional rule: (3)$\alpha \in \mathcal{S}$ implies $\Box^\alpha : \alpha$.

Let $\mathcal{R}$ be a TRS. A sort attachment τ is said to be *consistent* with $\mathcal{R}$ if for any rewrite rule $l \to r \in \mathcal{R}$, l and r are well-sorted under τ with the same sort. Note that $\mathcal{R}^\tau$ acts on $\mathcal{T}^\tau$, i.e. well-sorted terms $s, t \in \mathcal{T}^\tau$ whenever $s \to_{\mathcal{R}^\tau} t$; and that for any $s, t \in \mathcal{T}^\tau$, $s \to_{\mathcal{R}} t$ if and only if $s \to_{\mathcal{R}^\tau} t$.

From a given TRS $\mathcal{R}$ and a sort attachment τ consistent with $\mathcal{R}$, by regarding each function symbol f to be of sort $\tau(f)$ and each variable x to be of sort $\tau(x)$, we get a STRS $\mathcal{R}^\tau$ - called a STRS induced from $\mathcal{R}$ and τ.

Using the sort attachment, persistence can be alternatively formulated as follows. It is clear that definition of Zantema [13] and the following definition are equivalent.

Definition 2.1 A property P is called *persistent* if for any TRS $\mathcal{R}$ and any sort attachment τ that is consistent with $\mathcal{R}$ the following property holds:

$$\mathcal{R}^\tau \text{ has the property } P \Leftrightarrow \mathcal{R} \text{ has the property } P.$$

We consider the persistent property for TRSs using definition 2.1 in this paper instead of Zantema's definition. From now on, we assume that a set $\mathcal{S}$ of sorts, a TRS $\mathcal{R}$ are given. Then an attachment τ on $\mathcal{S}$ that is consistent with $\mathcal{R}$ is fixed.

3 Characterizations of well-sortedness

In this section we give a characterization of well-sortedness.

Definition 3.1 The *top sort* (under τ) of an unsorted term t is defined as follows:

$$top(t) = \begin{cases} \tau(t) & \text{if } t \in \mathcal{V} \\ \beta & \text{if } t = f(t_1, \ldots, t_n) \text{ with } \tau(f) = \alpha_1 \times \ldots \times \alpha_n \to \beta \end{cases}$$

Definition 3.2 Let $t = C[t_1, \dots, t_n]$ $(n \geq 0)$ be an unsorted terms with $C[, \dots ,] \neq \Box$. We write $t = C[\![t_1, \dots, t_n]\!]$ if and only if

(1) $C{:}\alpha_1 \times \dots \times \alpha_n \to \beta$ is a context that is well-sorted under τ.

(2) $top(t_i) \neq \alpha_i$ for $i = 1, \dots, n$.

The $t_1,\dots,t_n$ are said to be the *principal subterms* of t.

We denote $t = C\langle\langle t_1, \dots, t_n \rangle\rangle$ if either $t = C[\![t_1, \dots, t_n]\!]$ or $C = \Box$ and $t_i \in \{t_1,\dots,t_n\}$.

Definition 3.3 Let t be an unsorted term. The *rank* of t is defined by

$$rank(t) = \begin{cases} 1 & \text{if } t \text{ is well-sorted term} \\ 1 + max\{rank(t_1), \dots, rank(t_n)\} & \text{if } t = C[\![t_1, \dots, t_n]\!] \end{cases}$$

We consider the example of top sort, principal subterm and rank of an unsorted term.

Example 3.4 Let $\mathcal{F} = \{f, g, h, A, B\}$, $\mathcal{S} = \{0, 1\}$ and $\tau = \{f : 0 \times 0 \to 1, g : 1 \to 0, h : 0 \times 1 \times 1 \to 1, A : 0, B : 0\}$.
We consider the unsorted term $f(g(A), h(x, B, B))$.

- $top(f(g(A), h(x, B, B))) = 1$ because of $\tau(f) = 0 \times 0 \to 1$.
- $f(g(A), h(x, B, B)) = C[\![A, h(x, B, B)]\!]$ where $C[, \dots ,] = f(g(\Box), \Box)$. The principal subterms of $f(g(A), h(x, B, B))$ are A and $h(x, B, B)$.
- $rank(f(g(A), h(x, B, B))) = 1 + max\{rank(A), rank(h(x, B, B))\} = 3$.

Definition 3.5 A rewrite step $s \to_{\mathcal{R}} t$ is said to be *inner* (written as $s \to^{i}_{\mathcal{R}} t$) if and only if $s = C\ [\![s_1,\dots,C'\ [l\theta],\dots,s_n]\!] \to_{\mathcal{R}} C\ [\![s_1,\dots,C'\ [r\theta],\dots,s_n]\!] = t$ for some $s_1,\dots,s_n$, $l \to r \in \mathcal{R}$, θ and C', otherwise *outer* (written as $s \to^{o}_{\mathcal{R}} t$).

Next, we give the example of inner and outer rewrite step.

Example 3.6 Let $\mathcal{F} = \{f, g, h, G, A, B\}$, $\mathcal{S} = \{0, 1\}$ and $\tau = \{f : 0 \times 0 \to 1, G : 0 \to 1, g : 1 \to 0, h : 0 \times 1 \times 1 \to 1, A : 0, B : 0\}$. We consider the following TRS:

$$\mathcal{R} = \begin{cases} f(x, y) \to G(y) \\ h(x, z, z) \to z \end{cases}$$

The following reduction sequence starting from unsorted term $f(g(A),h(x, B, B))$:

$$f(g(A), \underline{h(x, B, B)}) \xrightarrow[\mathcal{R}]{i} \underline{f(g(A), B)} \xrightarrow[\mathcal{R}]{o} G(B)$$

4 Persistence of termination for non-overlapping TRSs

In this section we show the persistence of termination for non-overlapping TRSs. It is *main theorem* in this paper. First, we show the persistence of weak innermost normalization. Next, we show the persistence of termination for non-overlapping TRSs. Furthermore we give the simple example as application of our main result.

Let $s_1,\ldots,s_n$ and $t_1, \ldots, t_n$ be terms. We write $\langle s_1,\ldots,s_n \rangle \propto \langle t_1,\ldots,t_n \rangle$ if and only if for any $1 \leq i,j \leq n$, $s_i = s_j$ implies $t_i = t_j$. Moreover, we write $\langle s_1,\ldots,s_n \rangle \infty \langle t_1,\ldots,t_n \rangle$ if and only if $\langle s_1,\ldots,s_n \rangle \propto \langle t_1,\ldots,t_n \rangle$ and $\langle t_1,\ldots,t_n \rangle \propto \langle s_1,\ldots,s_n \rangle$.

The following theorem was proved by Gramlich in [6].

Theorem 4.1 *([6]) Let $\mathcal{R}$ be a non-overlapping TRS. Then, $\mathcal{R}$ is weakly innermost normalizing if and only if $\mathcal{R}$ is terminating.*

We give the proof of persistence of weak innermost normalization.

Theorem 4.2 *Weak innermost normalization is a persistent property of TRSs.*

Proof. Let $\mathcal{R}$ be a *TRS*. We show that $\mathcal{R}^\tau$ is weakly innermost normalizing if and only if $\mathcal{R}$ is weakly innermost normalizing.

- (if)-part: For well-sorted term $s,t \in \mathcal{T}^\tau$, $s \to_{\mathcal{R}^\tau} t$ if and only if $s \to_{\mathcal{R}} t$. Hence, every well-sorted term has a normal form which can be reached by an innermost reduction.

- (only if)-part: We will show by induction on $rank(t)$ that every unsorted term t has a normal form which can be reached by an innermost reduction with respect to $\mathcal{R}$. If $rank(t) = 1$ then the result follows from the assumption that $\mathcal{R}$ is weakly innermost normalizing. Let $t = C[\![t_1,\ldots,t_n]\!]$. Applying the induction hypothesis to $t_1,\ldots,t_n$ yields normal forms $t'_1,\ldots,t'_n$ such that $t_j \to^*_{i\mathcal{R}} t'_j$ for $j = 1, \ldots, n$. We clearly have $C[t'_1,\ldots,t'_n] = C'[\![s_1,\ldots,s_m]\!]$ for some context $C'[,\ldots,]{:}\alpha_1 \times \ldots \times \alpha_m \to \alpha$ and normal forms $s_1,\ldots,s_m$. Choose fresh variables $x_i \in \mathcal{V}^{\alpha_i}$ for $i = 1,\ldots,m$ such that $\langle x_1,\ldots,x_m \rangle \infty \langle s_1,\ldots,s_m \rangle$. Because $rank(C'[x_1,\ldots,x_m]) = 1$, the well-sorted term $C'[x_1,\ldots,x_m]$ has a normal form which can be reached by an innermost reduction, say $C'[x_1,\ldots,x_m] \to^*_{i\mathcal{R}} C^*[x_{i1},\ldots,x_{ip}]$. Hence, we have the following innermost reduction sequence:

  $t \to^{i*}_{i\mathcal{R}} C'[\![s_1,\ldots,s_m]\!] \to^{o*}_{i\mathcal{R}} C^*\langle\langle s_{i1},\ldots,s_{ip}\rangle\rangle = t'$. Clearly t' is normal form which can be reached by an innermost reduction with respect to $\mathcal{R}$. We conclude that every unsorted term has a normal form which can be reached by an innermost reduction with respect to $\mathcal{R}$. □

We obtain the main theorem in this paper form theorem 4.1 and theorem 4.2.

Theorem 4.3 *Termination is a persistent property of non-overlapping TRSs.*

Proof. Let $\mathcal{R}$ be a non-overlapping *TRS*. We have to show that $\mathcal{R}^\tau$ is terminating if and only if $\mathcal{R}$ is terminating. By theorem 4.1, $\mathcal{R}$ is weakly innermost normalizing if and only if $\mathcal{R}$ is terminating. By theorem 4.2, $\mathcal{R}^\tau$ is weakly innermost normalizing if and only if $\mathcal{R}$ is weakly innermost normalizing. Hence, $\mathcal{R}^\tau$ is weakly innermost normalizing if and only if $\mathcal{R}$ is terminating. Since TRS $\mathcal{R}$ is non-overlapping, so is STRS $\mathcal{R}^\tau$. By theorem 4.1, $\mathcal{R}^\tau$ is weakly innermost normalizing if and only if $\mathcal{R}^\tau$ is terminating. Therefore, $\mathcal{R}^\tau$ is terminating if and only if $\mathcal{R}$ is terminating. $\square$

Example 4.4 We show that the following non-overlapping TRS $\mathcal{R}$ is terminating using theorem 4.3. To show the termination of the following TRS directly seems difficult from known results (E.g. recursive path ordering [5]). Also, we can *not* use the modularity results for composable TRSs [9] and hierarchical combinations of TRSs [8]. Zantema's result that termination is persistent for TRSs without collapsing or duplicating rules can *not* be applied, because next TRS contains both collapsing rule (r2) and duplicating rule (r4). However, we can show the termination of next TRS using our results in this paper.

$$\mathcal{R} = \begin{cases} g(x,B) \to g(x,A) & (r1) \\ g(x,d(z,B)) \to x & (r2) \\ I(A,g(x,d(y,C))) \to I(B,g(x,d(y,C))) & (r3) \\ d(z,A) \to e(z,z) & (r4) \end{cases}$$

Let $\mathcal{S} = \{0,1,2\}$. We give the following sort attachment τ.

$$\tau = \begin{cases} g : 1 \times 0 \to 1 \\ I : 0 \times 1 \to 2 \\ d : 0 \times 0 \to 0 \\ e : 0 \times 0 \to 0 \\ A : 0, B : 0, C : 0 \end{cases}$$

Any well-sorted term in $\mathcal{T}^0$, $\mathcal{T}^1$ and $\mathcal{T}^2$ is terminating, i.e. any well-sorted term in $\mathcal{T}^\tau$ is terminating. We consider the following cases:

- $t \in \mathcal{T}^0$. Then $(r4)$ is the only applicable rule. A TRS $\{(r4)\}$ is terminating using recursive path ordering. Hence, t is terminating.
- $t \in \mathcal{T}^1$. Then $(r1)$, $(r2)$ and $(r4)$ are the only applicable rules. A TRS $\{(r1), (r2), (r4)\}$ is terminating using recursive path ordering. Hence, t is terminating.
- $t \in \mathcal{T}^2$. Then $(r1)$, $(r2)$, $(r3)$ and $(r4)$ are the applicable rules. If $top(t) \neq 2$, then t is terminating. If $top(t) = 2$, then we show that t is terminating, since the above two cases. For any proper subterm s of t, $top(s) = 0$ or $top(s) = 1$. Since the above two cases, s is terminating. Since $top(t) = 2$, $(r3)$ is the only applicable rule to root position of term t. Hence, t is terminating.

Then, STRS $\mathcal{R}^\tau$ is terminating. Since $\mathcal{R}^\tau$ is non-overlapping TRS and theorem 4.3, TRS $\mathcal{R}$ is terminating.

Furthermore we obtain the persistence of completeness for non-overlapping TRSs.
The following theorem was given by Aoto and Toyama [1].

Theorem 4.5 *([1]) Confluence is a persistent property of TRSs.*

Since a complete TRS is confluent and terminating, we obtain the following corollary from theorem 4.3 and theorem 4.5.

Corollary 4.6 *Completeness is a persistent property of non-overlapping TRSs.*

5 Conclusion

In this paper, we have discussed the persistence of termination for non-overlapping TRSs. We have given our main results in the following.

First, we have shown the persistence of weak innermost normalization. Next, we have shown the persistence of termination for non-overlapping TRSs and we have given the example as application of our main result. This result has provided a new and powerful tool for proving termination of TRSs. Furthermore we have obtained the persistence of completeness for non-overlapping TRSs.

Acknowledgments

The authors would like to thank Prof. Yoshihito Toyama, Prof. Masahiko Sakai, Dr. Keiichiro Kusakari and Dr. Takahito Aoto for their useful discussions, comments and suggestions.

References

[1] T. Aoto and Y. Toyama, "Persistency of confluence," J. Universal Computer Science, 3, pp.1134–1147, 1997.

[2] T. Aoto and Y. Toyama, "Extending persistency of confluence with ordered sorts," Research report, IS-RR-96-0025F, School of Information Science, JAIST, 1996.

[3] F. Baader and T. Nipkow, "Term rewriting and all that," Cambridge University Press, 1998.

[4] N. Dershowitz and J. P. Jouannaud, "Rewrite systems," In Handbook of Theoretical Computer Science, vol.B, ed. J.van Leeuwen, pp.243–320, The MIT Press/Elsevier, 1990.

[5] N. Dershowitz, "Termination of rewriting," J. Symbolic Computation, 3, pp.69–116, 1987.

[6] B. Gramlich, "Abstract relations between restricted termination and confluence properties of rewrite systems," Fundamenta Informaticae, 24, pp.3–23, 1995.

[7] J. W. Klop, "Term rewriting systems," In Handbook of Logic in Computer Science, vol.2, pp.1–112, ed. S. Abramsky, D. Gabbay and T. Mabiaum, Oxford University Press, 1992.

[8] M.R.K. Krishna Rao, "Modular proofs for completeness of hierarchical term rewriting systems," Theoretical Computer Science, 151, pp.487–512, 1995.

[9] E. Ohlebusch, "Modular properties of composable term rewriting systems," J. Symbolic Computation, 20 (1), pp.1–41, 1995.

[10] E. Ohlebusch, "Advanced topics in term rewriting," Springer-Verlag, 2002.

[11] H. Ohsaki and A. Middeldorp, "Type introduction for equational rewriting," Proc. 4th Symp. on Logical Foundations of Computer Science, LNCS, 1234, pp.283–293, 1997.

[12] Y. Toyama, "Counterexamples to termination for the direct sum of term rewriting systems," Information Processing Letters, 25, pp.141–143, 1987.

[13] H. Zantema, "Termination of term rewriting: interpretation and type elimination," J. Symbolic Computation, 17, pp.23–50, 1994.